#ODSC

ODSC West 2017

Monitoring Spacecraft Telemetry with a Fleet of LSTMs

Kyle Hundman
Data Scientist, Group Lead

11.3.17

Jet Propulsion Laboratory
California Institute of Technology

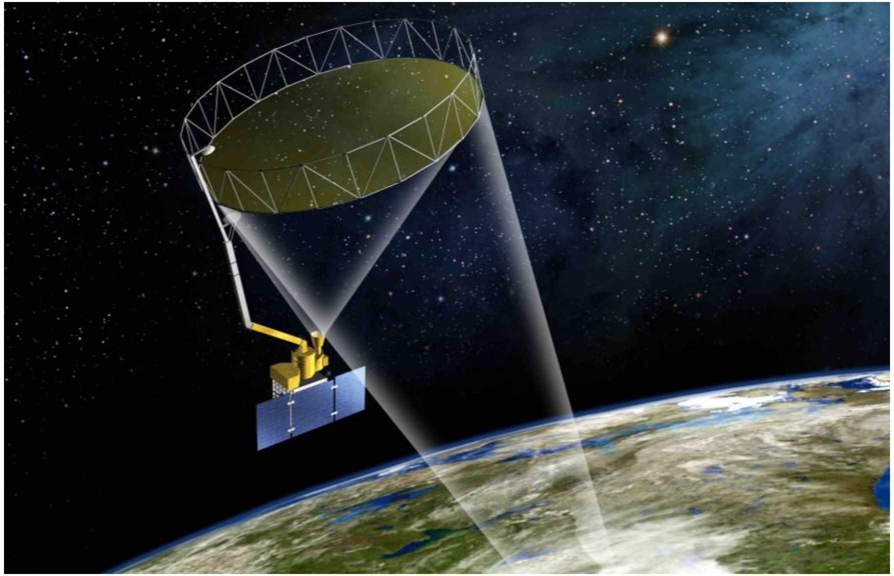
1. **Origins and Motivation**
2. **Anomaly Detection and LSTMs**
3. **Implementation and Proof of Concept**
4. **As a System**
5. **Wrap-Up**

Origins and Motivation

Megasystems of Sensors

Soil Moisture Active Passive (SMAP)

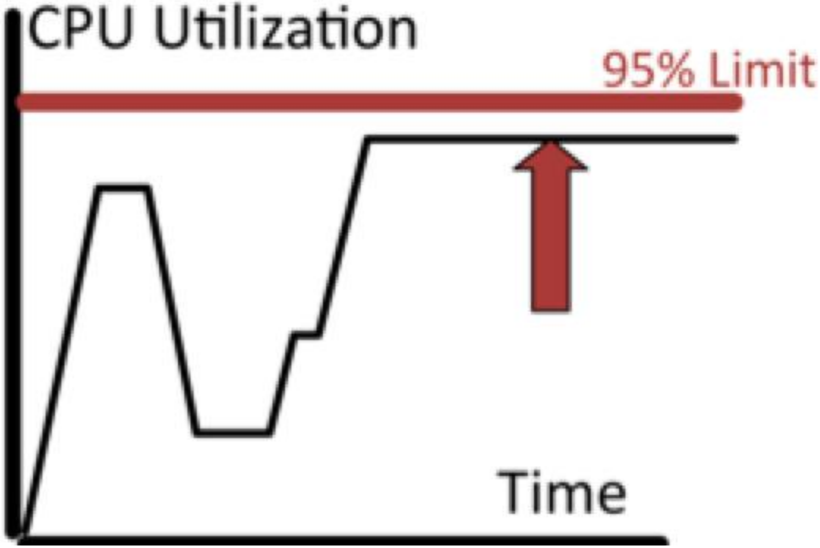
- ~4,000 telemetry channels
  - Power, CPU, RAM, Thermal, Radiation, counters, switches
- 4B values
- Challenges
  - Semi-supervised
  - Complexity, diversity
  - Scale vs. interpretability



Current System

Limit-checking and Expert System

- Engineers embed their knowledge and create alarms
  - Reliance on grey beards
  - Custom
  - Not complete
  - Accuracy
  - Appropriate limits change



Gathering Support

“How” not “why”?

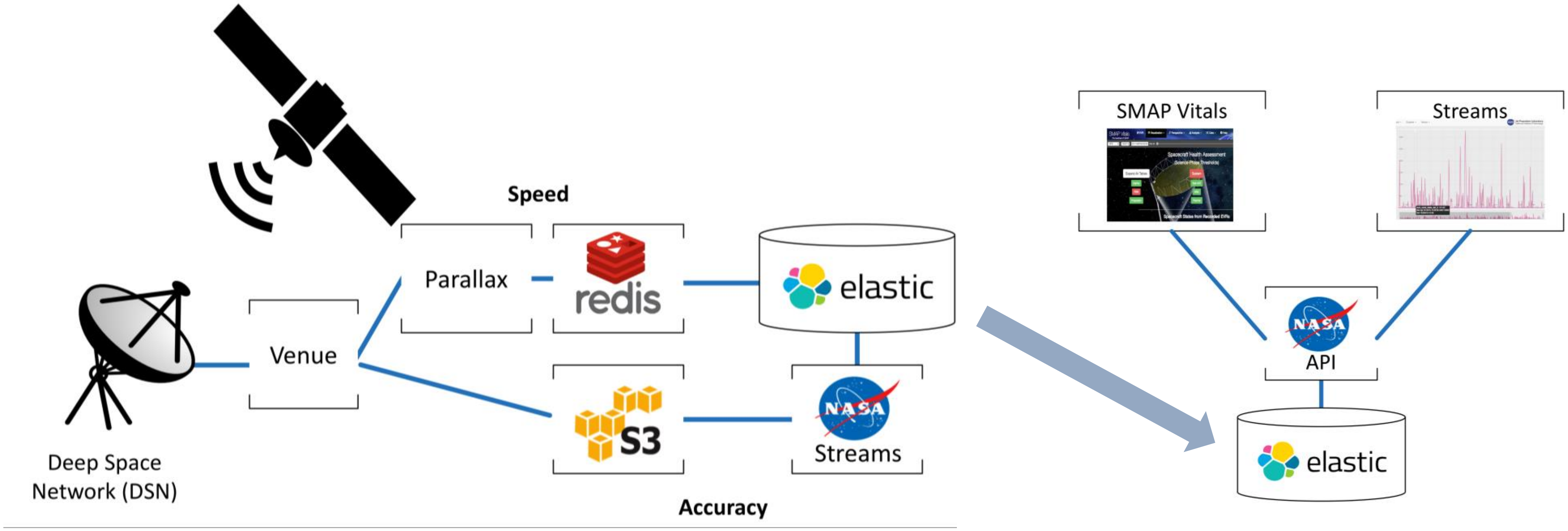
- In favor of
  - Harsh environment
  - Repairs are difficult
  - Risk Aversion
  - Generalization
  - Data infrastructure
- Against
  - Skepticism
  - Conservative mindset

SCIENCE NEWS | Thu Sep 3, 2015 | 4:51pm EDT

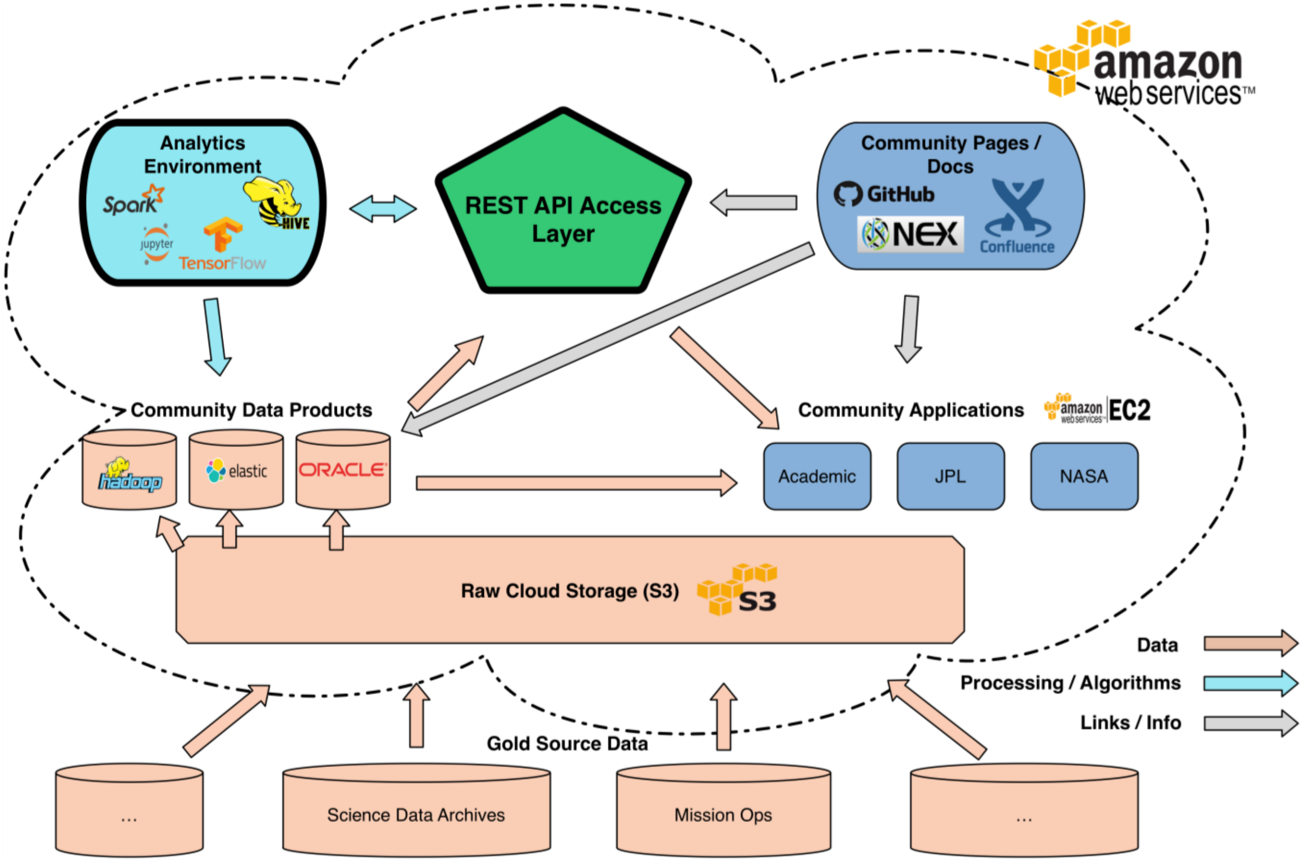
Key radar fails on $1 billion NASA environmental satellite

The "Analytics Cloud"

Data Engineering to enable Data Science



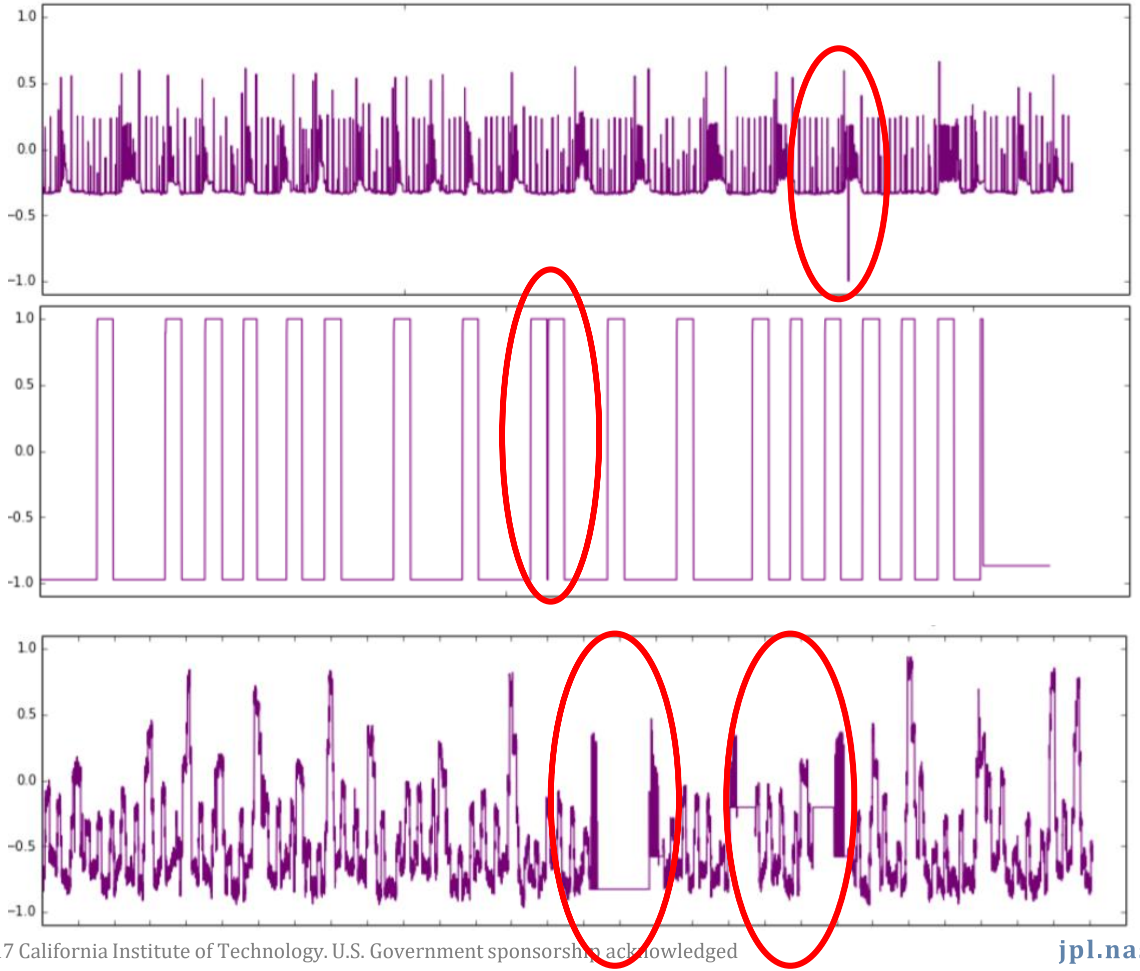
Bigger Picture – The Foundation



Anomaly Detection and Long Short-Term Memory Neural Nets (LSTMs)

Types of Anomalies

- Point
- Contextual
- Collective (sequential)



Anomaly Detection Survey

(cite)

Techniques	Classification Based Clustering Based Nearest Neighbor Based Statistical Information Theoretic Spectral
Applications	Cyber-Intrusion Detection Fraud Detection Medical Anomaly Detection Industrial Damage Detection Image Processing Textual Anomaly Detection Sensor Networks

Technique Used
Parametric Statistical Modeling
Non-parametric Statistical Modeling
Neural Networks
Spectral
Rule Based Systems

Technique Used
Bayesian Networks
Rule-based Systems
Parametric Statistical Modeling
Nearest Neighbor based Techniques
Spectral

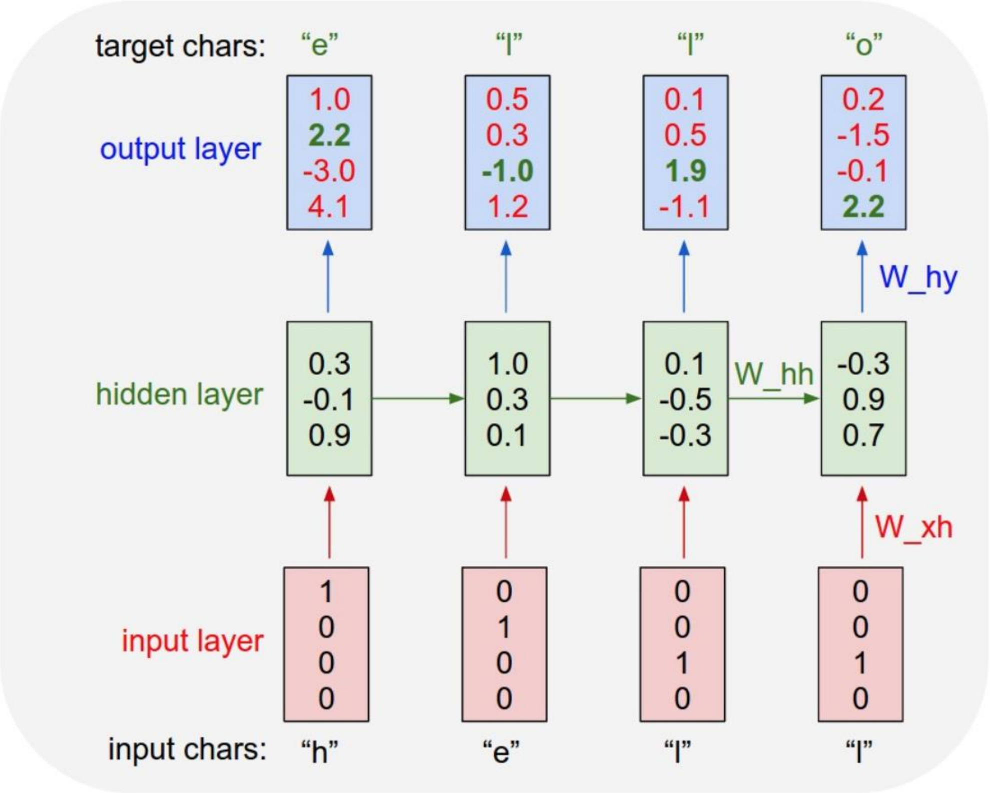
Disadvantages

Technique Family	Disadvantages
Classification	• acquiring labels (multi-class), • complexity
Nearest Neighbor	• misleading “neighborhoods” • choosing distance measure • complexity
Clustering	• difficulty of capturing cluster structure • complexity • distance measures • anomalies can form clusters
Statistical	• distribution assumptions (parametric) • lack of context (non-parametric, e.g. histograms)
Spectral	• High-computation complexity • Anomalies must be seperable in low-dimensional space

Point
Contextual ??
Collective ??

Recurrent Neural Nets

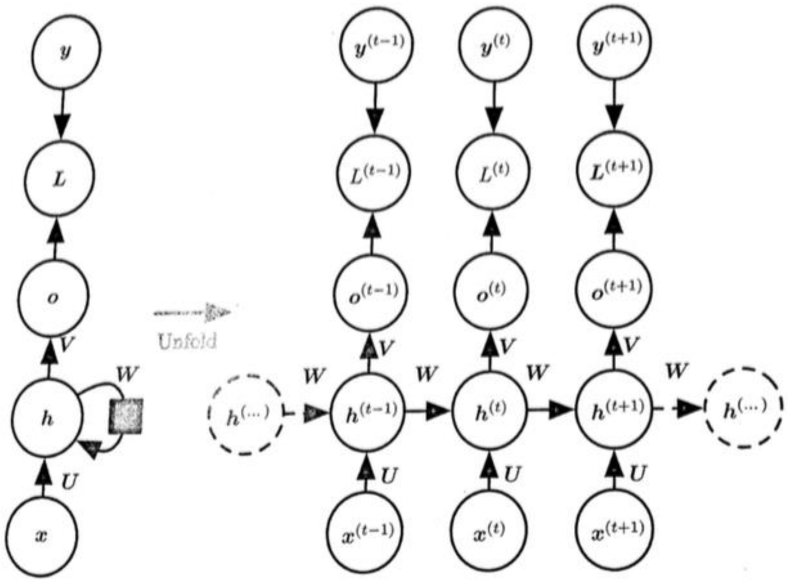
- Parameter sharing
  - Extend model to apply to different lengths and generalize across time steps
    - Don't have to have separate parameters for each time value
    - Share statistical learning across time (pieces of information are often recurring)
- Recurrence
  - Always has same input size regardless of sequence length



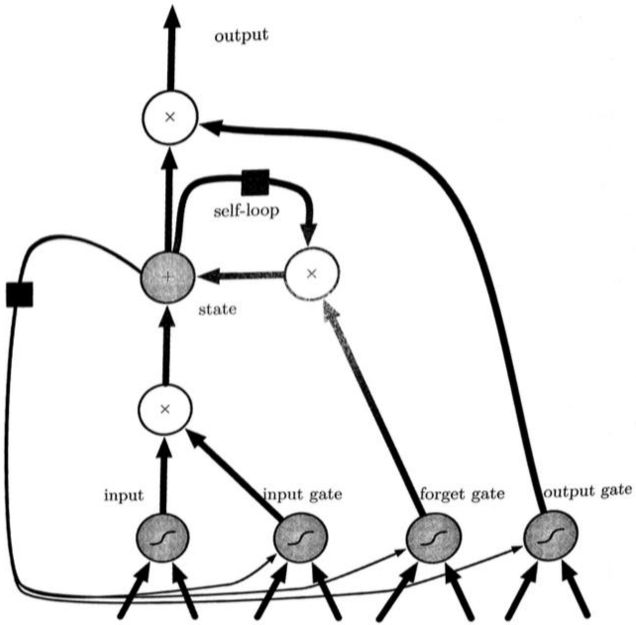
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

From RNNs to LSTMs (cite)

RNN



LSTM



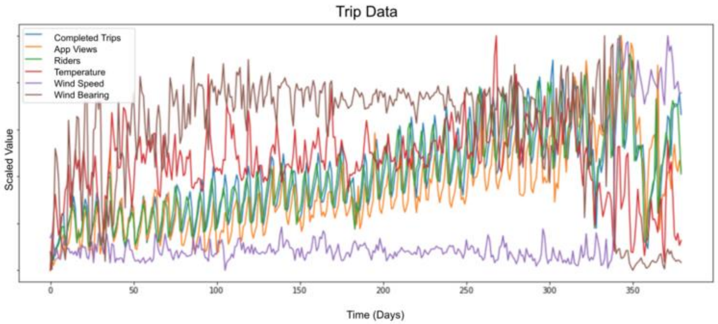
Core contribution (1997): Self-loops

Crucial addition (2000): Condition loop on context (with another hidden unit)

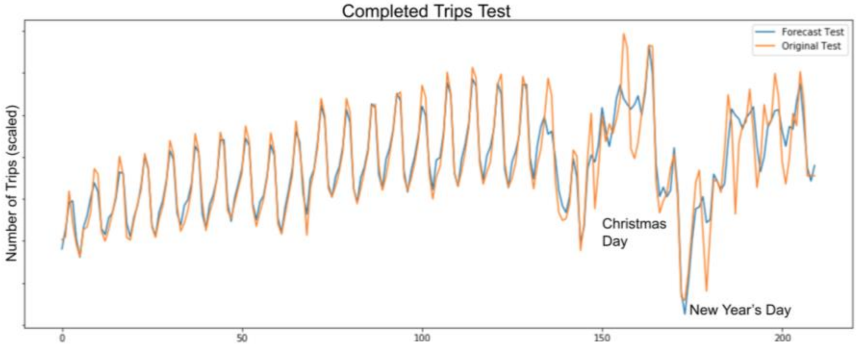
Implementation and Proof of Concept

Formulation

Many inputs → Single model → **Single output**



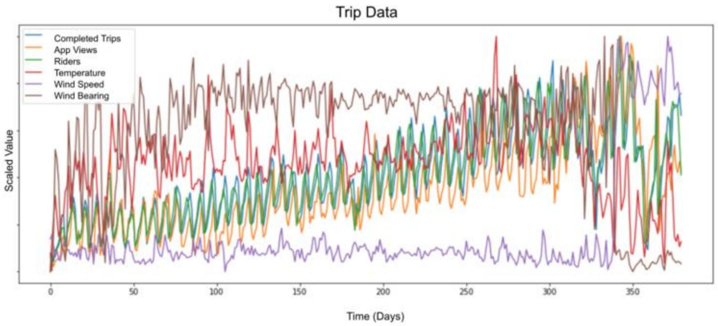
Model



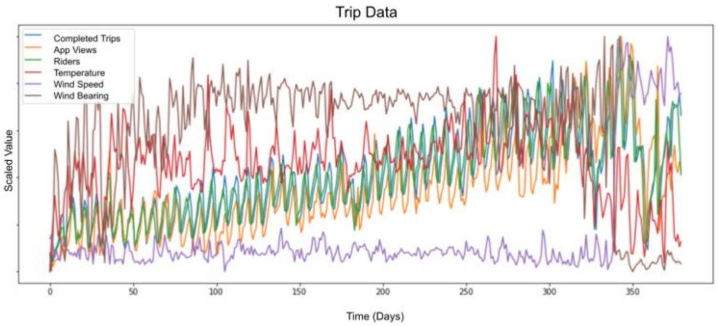
https://eng.uber.com/neural-networks/

Formulation

Many inputs → Single model → **Many outputs**

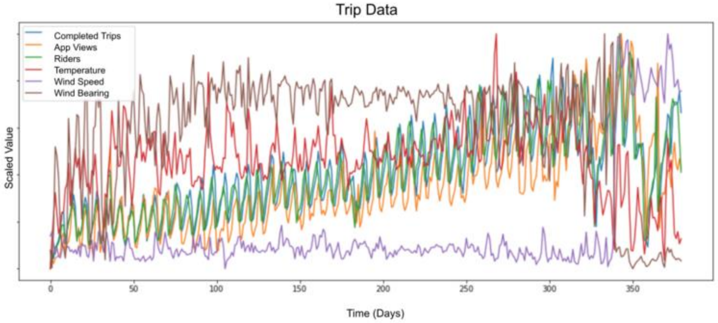


Model



Formulation

Many inputs → **Many models** → Many outputs

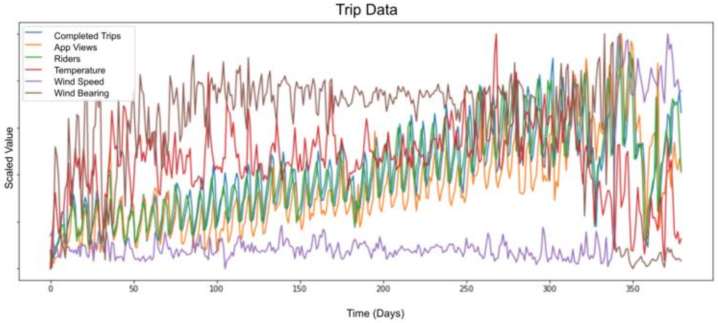


Model

Model

Model

Model



Why?

Granularity, control

Multi-dimensionality

Setup

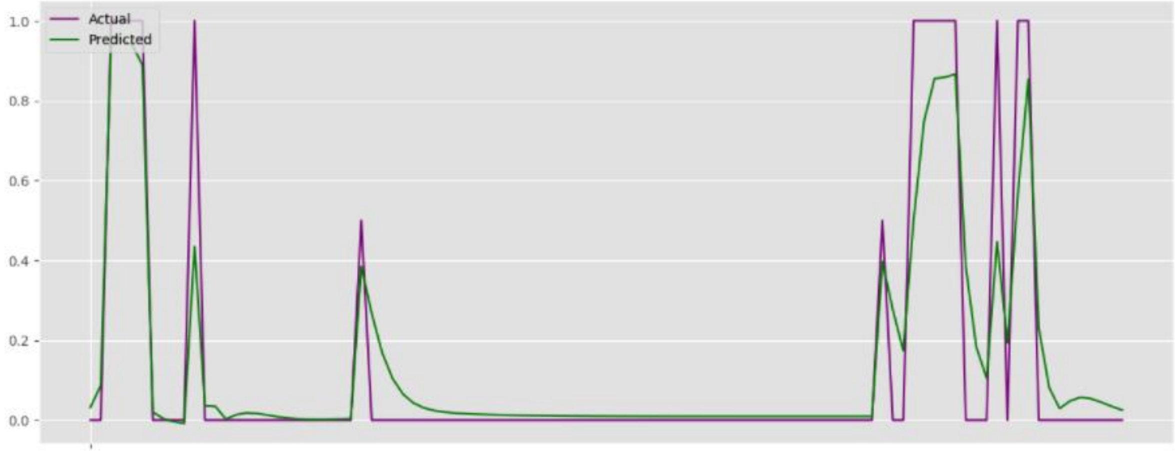
Time window	...	1	2	3	4	5	6	7	8	9	10	...
Channel values	...	65.3	66.4	63.9	74.1	75.3	72.9	65.7	66.1	66.4	...	
	...	66.4	63.9	74.1	75.3	72.9	65.7	66.1	66.2	???	...	

prediction

Recurrence

actual value

Errors 0.2 ...

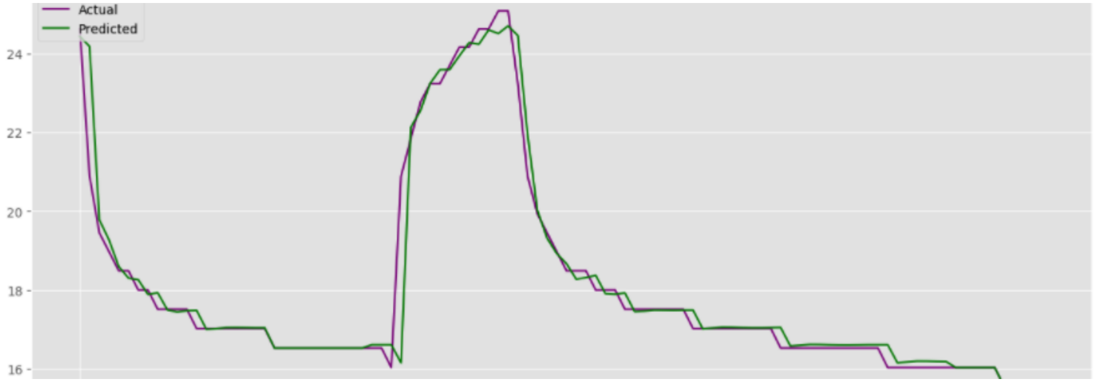


"Long Short Term Memory Networks for Anomaly Detection in Time Series"

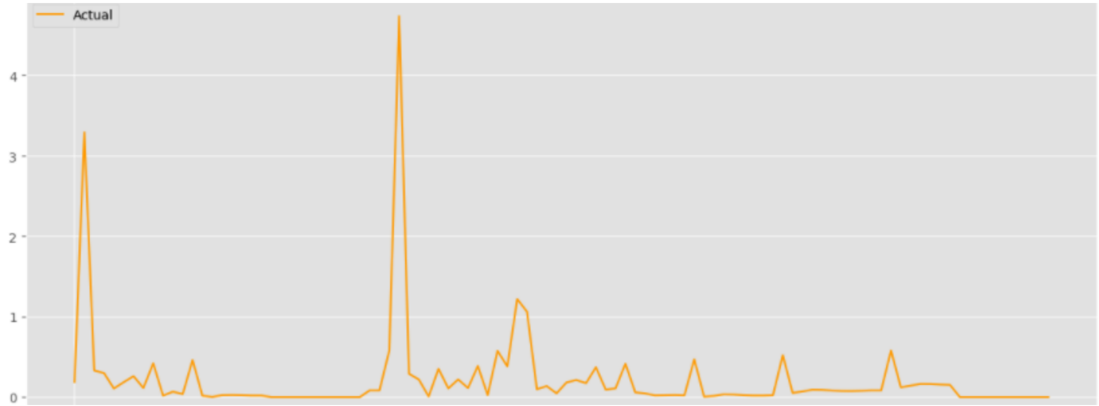
- Maholtra, , et. al (2015)

Reconstruction Errors and Smoothing

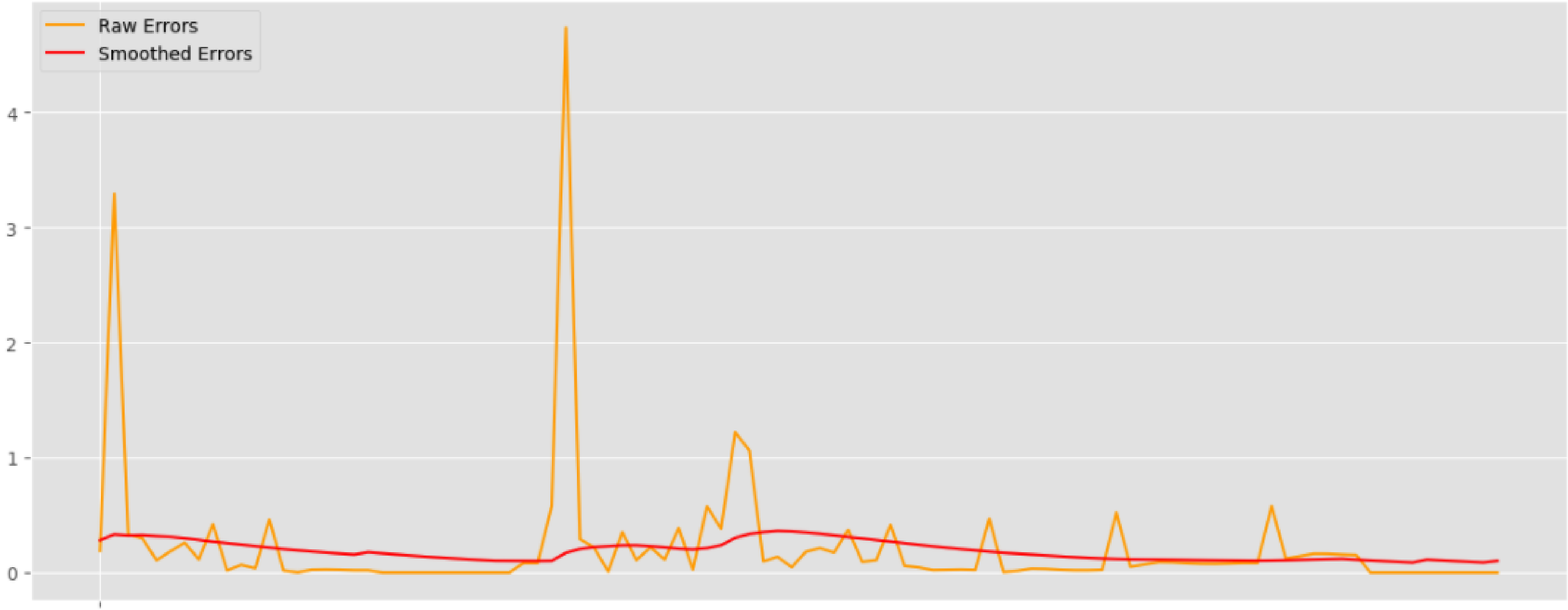
Actuals and Prediction



Raw Reconstruction Error

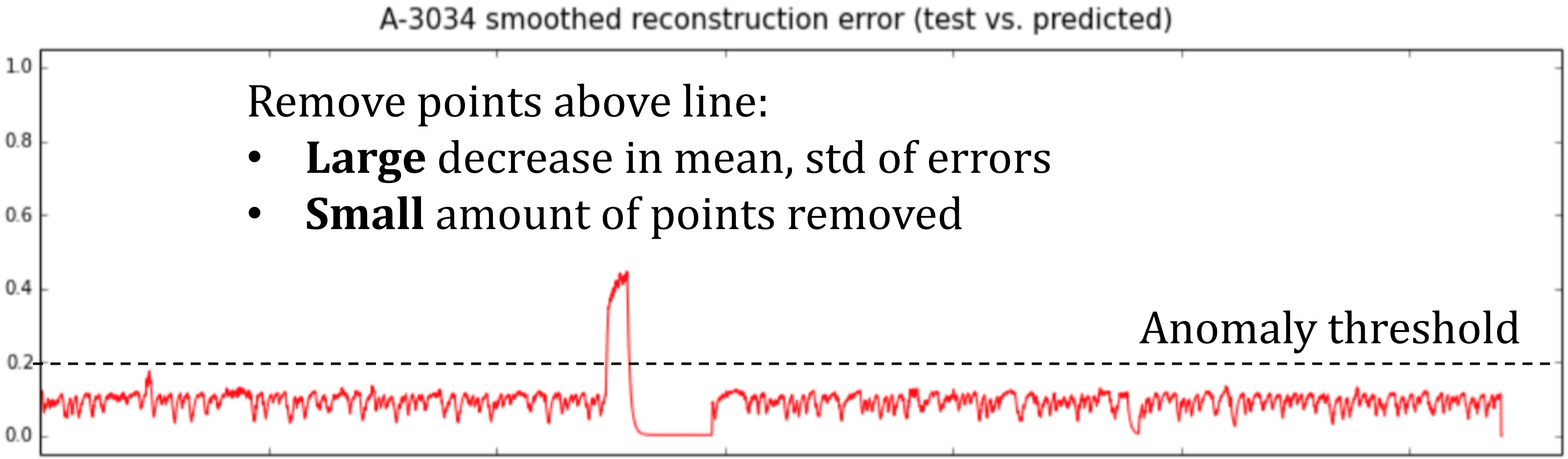


Reconstruction Errors and Smoothing



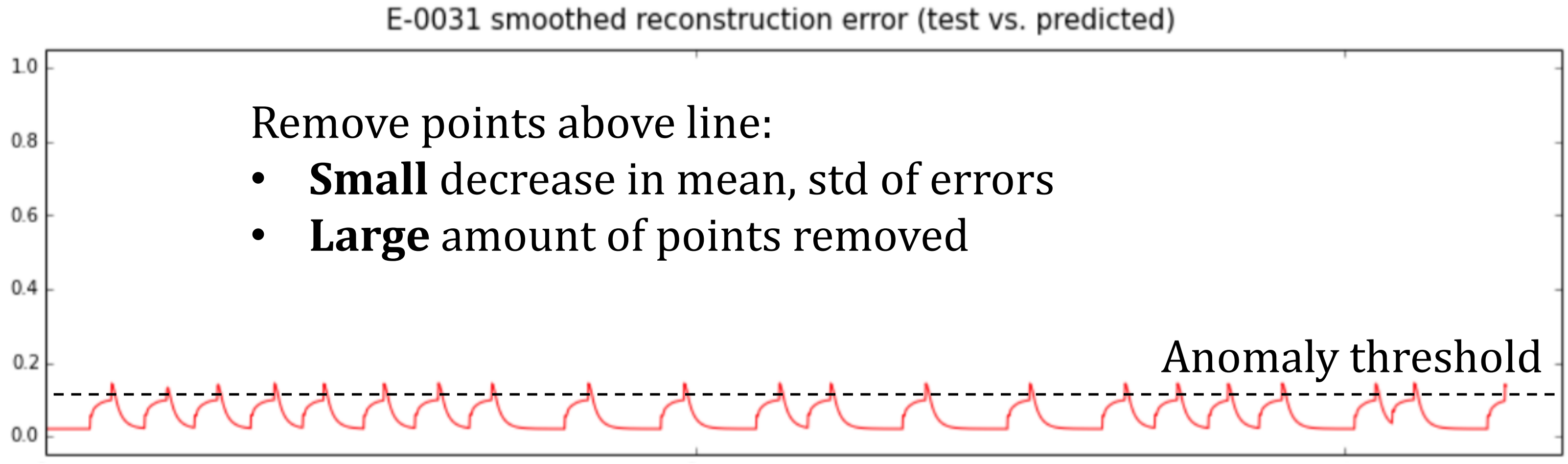
Dynamic Anomaly Threshold

Anomalous



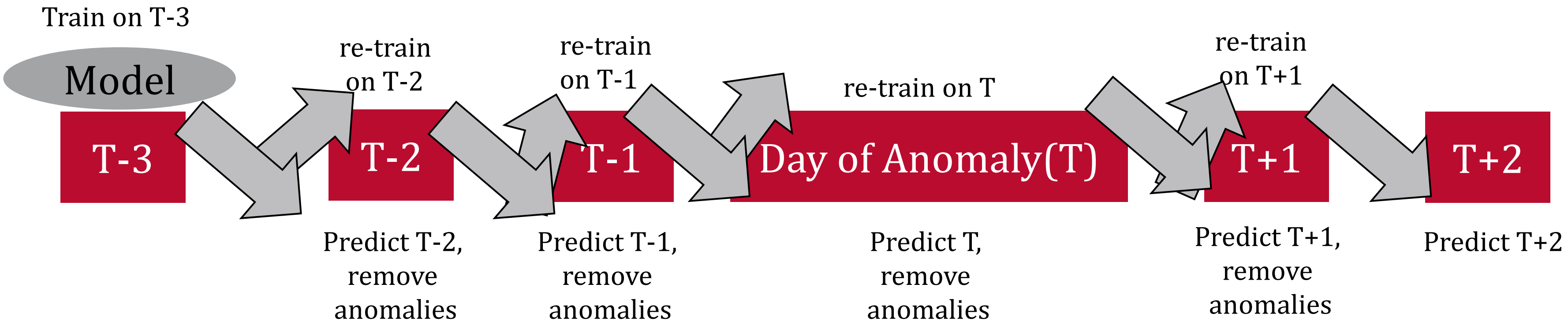
Dynamic Anomaly Threshold

Nominal



Experimentation – Incident Surprise, Anomaly Reports (ISAs)

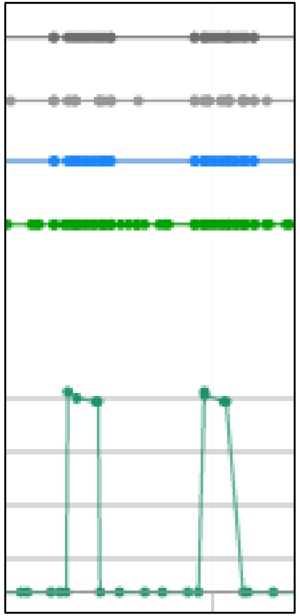
- Scraped ISAs to find mentions of telemetry channels and times (~130)
- “Turn on” 2 days before each anomaly, run through 2 days after
- Model trains on prior day, predicts current day



Experimental Results

	Pre-Anomaly	Day of Anomaly	Post-Anomaly	Total
TP	26	**65**	40	**131**
FP	51	**3**	28	**82**
FN	3	**8**	6	**17**
Precision	**34%**	**96%**	59%	**62%**
Recall	90%	**89%**	87%	**89%**

Incorporating Commands



Command/Diagnostic Activity

Module, Type, Instrument, Description

Telemetry Channel

Time

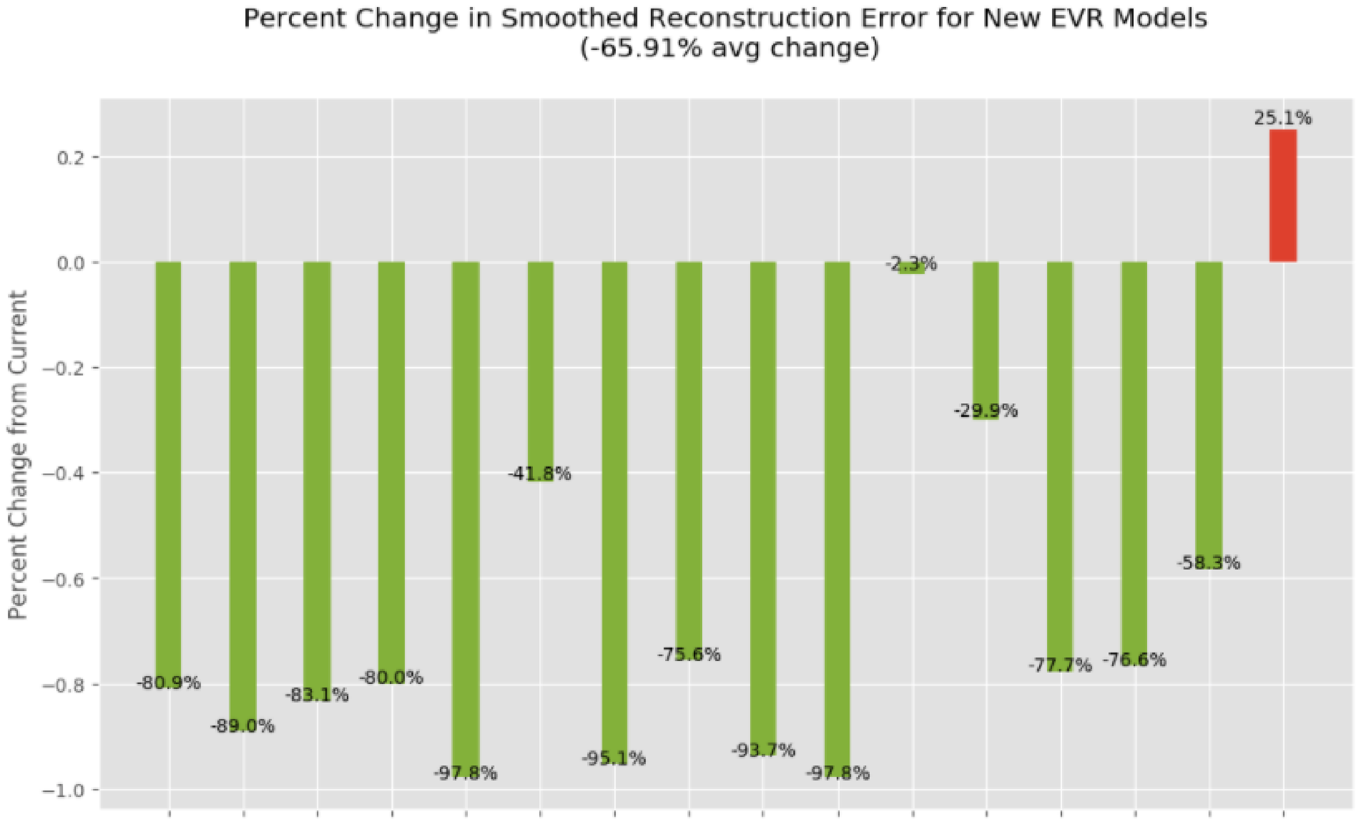
Using prior values

$$X_t = \begin{matrix} [\,[0.7]\,, \\ [0.4]\,, \\ [0.8]\,, \\ [0.2]\,] \end{matrix}$$

Using prior values with commands

$$X_t = \begin{matrix} [\,[0.7, 0, 0, 1, 0]\,, \\ [0.4, 0, 0, 0, 0]\,, \\ [0.8, 1, 0, 0, 0]\,, \\ [0.2, 0, 0, 0, 0]\,] \end{matrix}$$

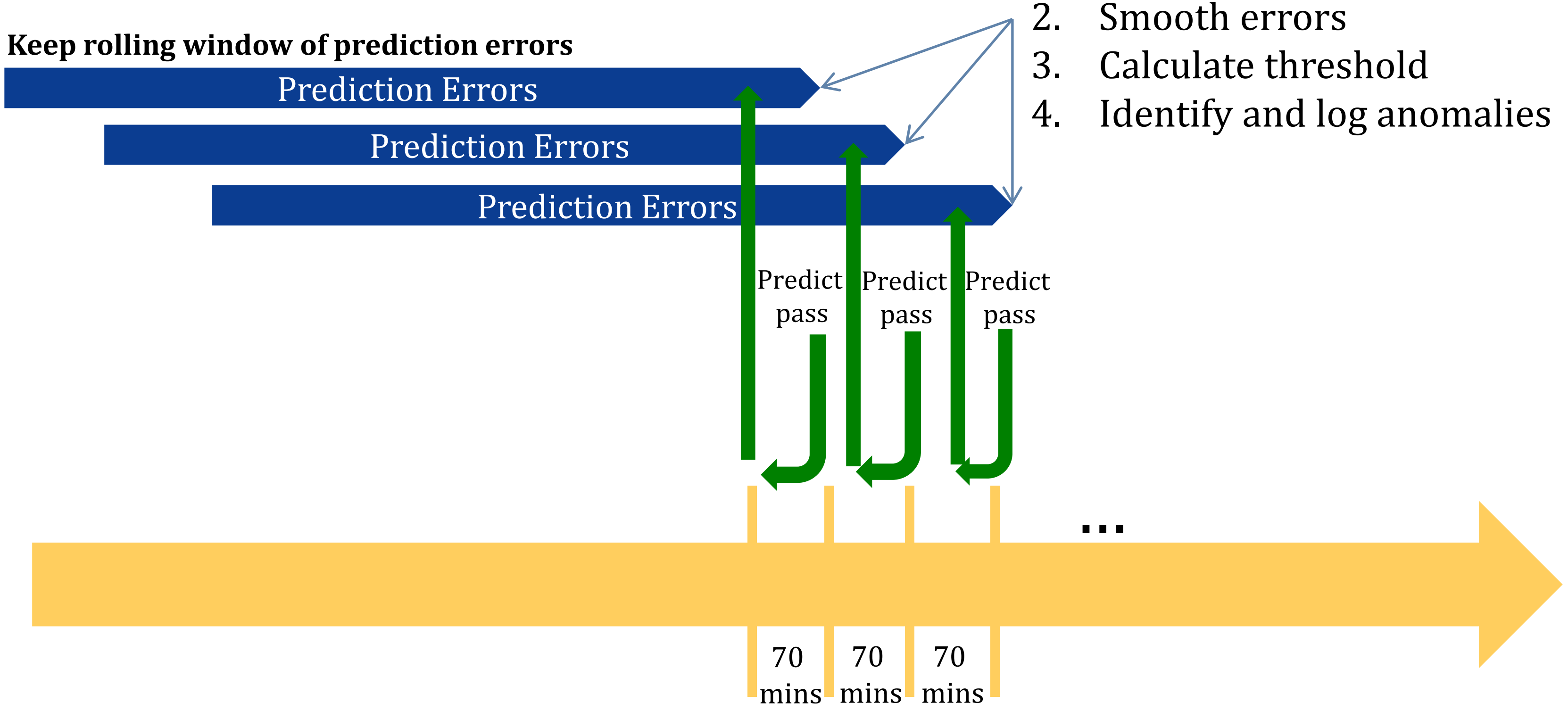
Incorporating Commands



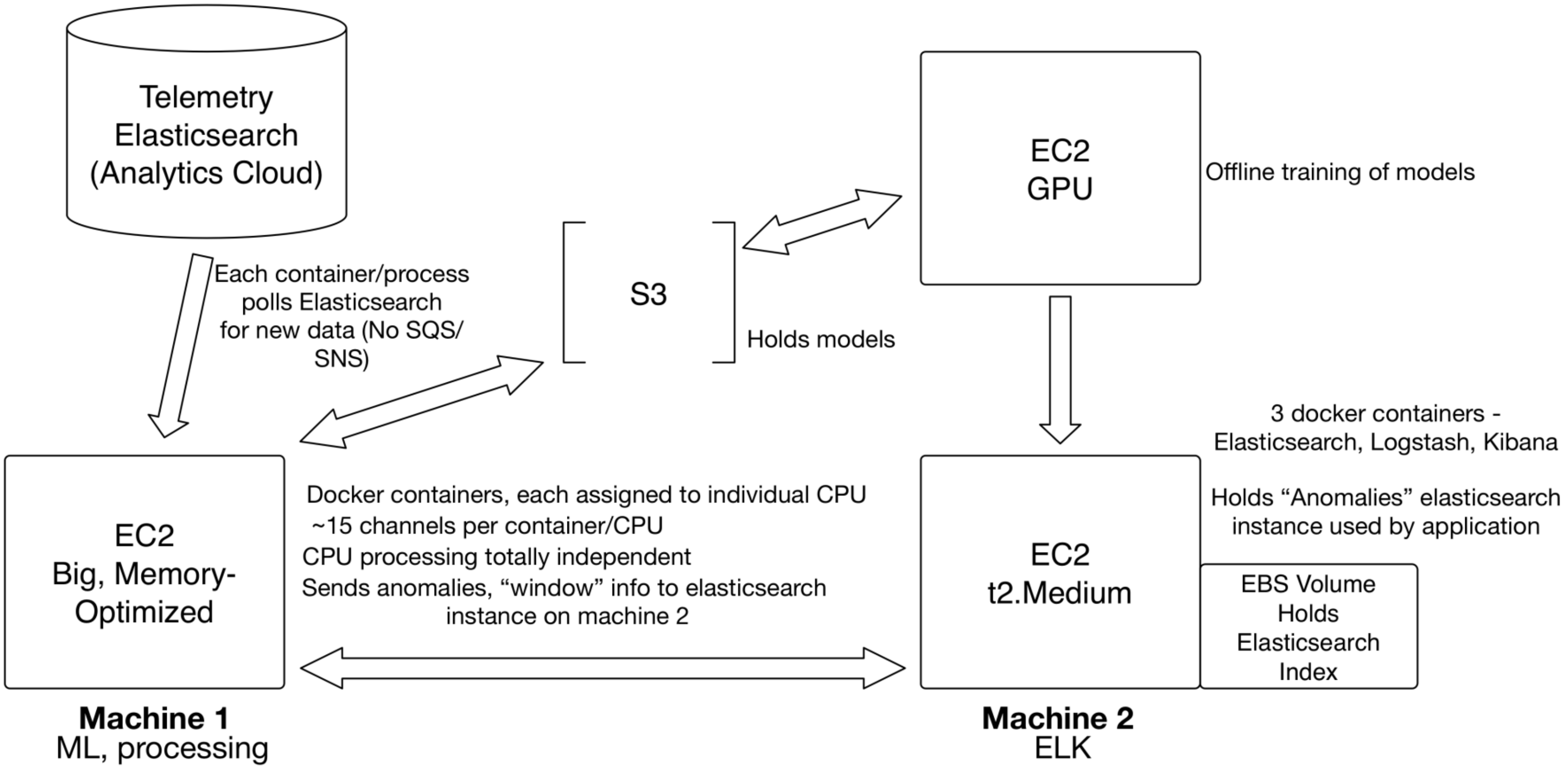
As a System

Processing (for each channel)

1. Predict new data and calculate errors across window
2. Smooth errors
3. Calculate threshold
4. Identify and log anomalies



Architecture

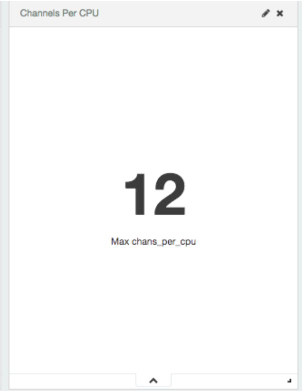


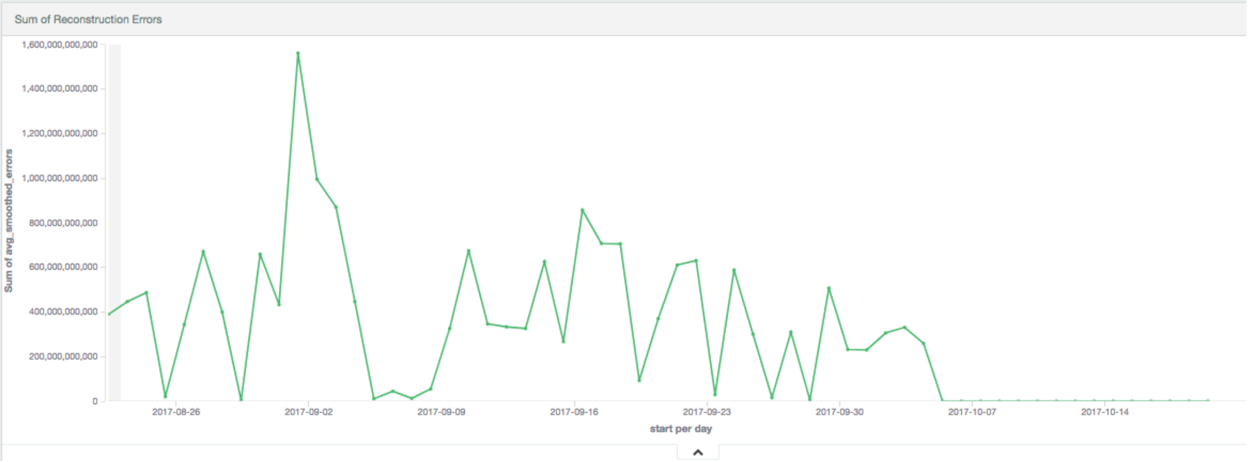
Rules of ML

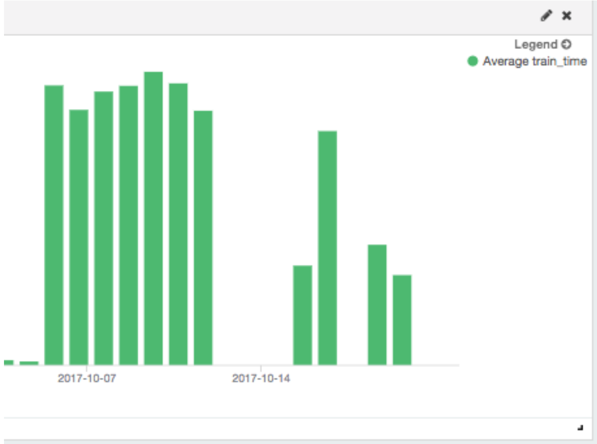
Martin Zenkivich (http://martin.zinkevich.org/rules_of_ml/rules_of_ml.pdf)

- "**Most of the problems you will face are engineering problems**. Even with all the resources of a great machine learning expert, **most of the gains come from great features**, not great machine learning algorithms. So, the basic approach is:
  - 1. make sure your pipeline is solid end to end
  - 2. start with a reasonable objective
  - 3. add commonsense features in a simple way
  - 4. make sure that your pipeline stays solid.
- This approach will make lots of money and/or make lots of people happy for a long period of time. Diverge from this approach only when there are no more simple tricks to get you any farther. **Adding complexity slows future releases**."

Monitoring the Monitoring System







Wrap-Up

Lessons and Considerations

- Good foundation in place (ETL could have been a lot harder)
- Clear benefit
- Big jump from proof of concept to system
- Can’t have too many monitoring and debugging capabilities
- RNNs are really impressive, toolkits are getting better

Future Work

- Interface
  - See to believe
- Generalizability, portability, robustness
  - “Once you've exhausted the simple tricks, cuttingedge machine learning might indeed be in your future.”
- Phased LSTMs
  - Time between data points
- Streaming, real-time implementation
  - Speed-ups: MXNet, compiled language processing
- Relationships
  - Anomaly correlations

Jet Propulsion Laboratory
California Institute of Technology

jpl.nasa.gov